

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-091257

(43)Date of publication of application : 04.04.1997

(51)Int.Cl.

G06F 15/16  
G06F 9/46

(21)Application number : 07-270513

(71)Applicant : NEC CORP

(22)Date of filing : 25.09.1995

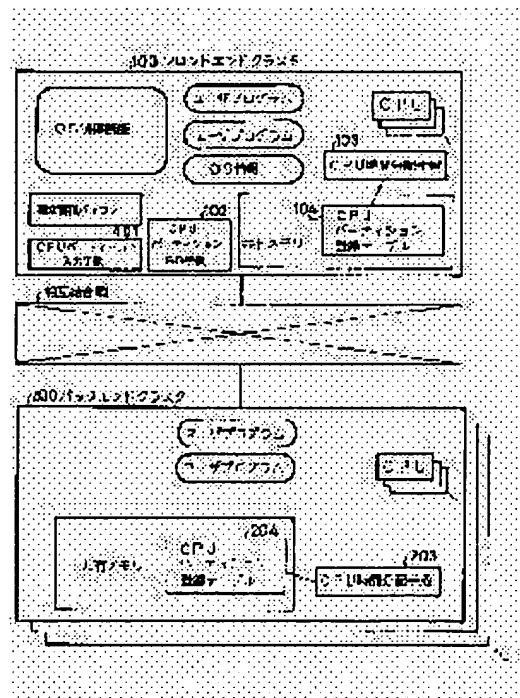
(72)Inventor : TANAKA HIDETOSHI

## (54) CPU MANAGEMENT SYSTEM

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To optimize the use of CPUs by dividing CPUs into groups to manage them as CPU partitions in a multi-cluster system.

**SOLUTION:** A CPU partition input means 101 inputs CPU partition information related to respective clusters (a front end cluster 100 and back end clusters 200). A CPU partition SG means 102 divides CPU groups of respective clusters into groups based on CPU partition information and registers information indicating group division in CPU partition registration tables 104 and 204 in each cluster. At the time of the occurrence of a process, a CPU time distribution means 103 or 203 assigns the CPU time related to CPUs belonging to the CPU partition, which is adapted to the character of the process, to the process based on reference to the CPU partition registration table 104 or 204.



## LEGAL STATUS

[Date of request for examination] 25.09.1995

[Date of sending the examiner's decision of rejection] 14.10.1998

[Kind of final disposal of application other than the examiner's decision of rejection or

application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's  
decision of rejection]

[Date of requesting appeal against examiner's  
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

**\* NOTICES \***

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

**DETAILED DESCRIPTION**

---

**[Detailed Description of the Invention]**

[0001]

[The technical field to which invention belongs] this invention relates to the CPU (Central Processing Unit) management method in a multi-cluster system.

[0002] Here, the "multi-cluster system" as used in this application specification means the system which has the feature as shown in following \*\* - \*\*.

\*\* It is constituted by two or more clusters (independent unit which consists of two or more CPUs and those shared memories).

\*\* It has OS (Operating System) control function, OS information, and a user program (these information is stored in the shared memory), and it has the demand accumulation buffer which accumulates the demand from a back-end cluster, and has the front end cluster which can exist only one in multi-cluster environment.

\*\* Without having OS control function, it has a user program (the information on a user program is stored in the shared memory), and has two or more back-end clusters which can exist by multi-cluster environment.

\*\* Two or more clusters (one front end cluster and one or more back-end clusters) are connected with the cross coupling network (generally high-speed network).

[0003]

[Description of the Prior Art] Before, in the computer system which has two or more CPUs, grouping of the CPU was carried out and the CPU management method which manages CPU based on the information about a group existed (for example, refer to JP, 1-109464, A).

[0004] However, by the conventional CPU management method, it was not able to be coped with about the above multi-cluster systems (in above-mentioned JP, 1-109464, A, reference is not made about a multi-cluster system). That is, the view of grouping was not introduced by the conventional CPU management method in a multi-cluster system.

[0005]

[Problem(s) to be Solved by the Invention] As mentioned above, by the conventional CPU management method, it could not be coped with suitable for a multi-cluster system, but there was a trouble that the effect (improvement in various performances) by carrying out grouping of the CPU was unrealizable in a multi-cluster system.

[0006] The purpose of this invention is set to a multi-cluster system in view of an above-mentioned point. By carrying out the grouping (grouping) of the CPU to every purpose of use (the content of a job), and managing as a CPU partition Optimization of use of CPU is attained. In the multi-cluster system concerned The various performances by the grouping of CPU (TAT of the responsibility of TSS (Time Sharing System) in a multi-cluster system, the throughput of a multiplex batch job, and the job about a parallel program ()) [ Turn ] Around It is in offering the CPU management method which can attain improvement in performances, such as Time.

[0007]

[Means for Solving the Problem] In the CPU management method in the multi-cluster system which connected two or more clusters in which the CPU management method of this invention contains one front end cluster with the cross coupling network The CPU partition input means in the front end cluster which inputs the CPU partition information about each cluster, A group division of two or more CPUs which can be set to each cluster based on the CPU partition information that it was inputted by the aforementioned CPU partition input means is performed. The CPU partition SG means in the front end cluster which registers into the CPU partition registration table in each cluster the information which shows the group division, It has a CPU-time distribution means in each cluster which assigns the CPU time concerning CPU belonging to the CPU partition which suits the character of the process concerned based on reference of the aforementioned CPU partition registration table at the time of generating of a process to the process concerned.

[0008]

[Embodiments of the Invention] Next, this invention is explained in detail with reference to a drawing.

[0009] The 1st example (example corresponding to invention according to claim 1) of the CPU management method of this invention is explained [ 1st ].

[0010] Drawing 1 is the block diagram showing the composition of the 1st example of the CPU management method of this invention.

[0011] The CPU management method of this example is realized in the multi-cluster system to which one front end cluster 100 and two or more back-end clusters 200 are connected with the cross coupling network (high-speed network).

[0012] The front end cluster 100 is constituted including the CPU partition input means 101, the CPU partition SG (System Generation) means 102, the CPU-time distribution means 103, and the CPU partition registration table 104 (it exists in a shared memory), after having OS control function, OS information and a user program (these information is stored in the shared memory although not illustrated), a demand accumulation buffer and two or more CPUs, and a shared memory.

[0013] The back-end cluster 200 is constituted including the CPU-time distribution means 203 and the CPU partition registration table 204 (it exists in a shared memory), after having two or more CPUs, a user program, and a shared memory.

[0014] In each cluster in a multi-cluster system as shown in drawing 1 , the CPU management method of this example carries out the grouping of two or more CPUs to every purpose of use (the content of a job), and manages them as a CPU partition. That is, it enables it to manage a CPU partition to specific every purpose of use (the content of a job) by registering into the CPU partition registration tables 104 or 204 corresponding to each cluster the information which carries out the grouping of two or more CPUs in each cluster (respectively [ the front end cluster 100 and the back-end cluster 200 ]), and shows the group division.

[0015] Drawing 2 is drawing for explaining concrete operation of the CPU management method of this example.

[0016] Drawing 3 is the flow chart showing processing of the CPU management method of this example. This processing consists of the CPU partition information input step 301, the CPU partition registration table creation step 302, the process generating judging step 303, a CPU selection step 304, and a CPU-time allocation step 305.

[0017] Next, operation of the CPU management method of this example constituted in this way is explained.

[0018] The CPU partition input means 101 in the front end cluster 100 inputs the CPU partition information which is information for characterizing a CPU partition and performing the management based on the directions from a system administrator about each cluster (respectively [ the front end cluster 100 and the back-end cluster 200 ]) (Step 301). Here, specifically, the CPU partition information in this example means the correspondence information on the CPU partition classified according to the contents of a job (a CPU bound (bound) job, an interactive (interactive) job, batch job, etc.), and the number of CPU belonging to the CPU partition.

[0019] The CPU partition SG means 102 creates the CPU partition registration tables 104 and 204 in

each cluster based on the CPU partition information that it was inputted at Step 301 (Step 302).

[0020] On the CPU partition registration table 104 in the example of drawing 2, CPU "3", "5", and "12" (CPU whose identification information of CPU is "3", "5", and "12") are made into one group, and those identification information is stored as information corresponding to a CPU partition "1" (CPU bound partition). Moreover, CPU "17", "18", "19", and "20" are made into one group, and those identification information is stored as information corresponding to a CPU partition "2" (interactive partition). Furthermore, CPU "7", "9", "22", and "23" are made into one group, and those identification information is stored as information corresponding to a CPU partition "3" (batch job partition). Storing (registration) of such information is performed not only about the CPU partition registration table 104 but about the CPU partition registration table 204 in each back-end cluster 200. By this, grouping of the identification information of CPU will be statically carried out to every [ of CPU ] purpose of use (the contents of a job).

[0021] If a process arises at the time of execution of the multi-cluster system concerned (step 303 reference), as the CPU-time distribution means 103 in the front end cluster 100 or the CPU-time distribution means 203 in each back-end cluster 200 shows allocation of the CPU time to the process to following \*\* and \*\*, it will perform it.

[0022] \*\* Choose one of the empty CPUs belonging to the CPU partition which suited the character (the purpose of use, i.e., the contents of a job about the process concerned) of the process for allocation based on the information about the CPU partition in the CPU partition registration table 104 (Step 304).

[0023] The CPU time which assigns CPU chosen by \*\* \*\* to the process fixed time, namely, starts the CPU is assigned to the process (Step 305), and it returns to the state of judging Step 303 after that.

[0024] The grouping of many CPUs in the multi-cluster system concerned can be carried out as mentioned above, and it can manage now as a CPU partition.

[0025] By introducing the view of such a CPU partition (grouping of CPU), optimization of distribution of the CPU time can be attained about performances, such as TAT of the various performances in a multi-cluster system, i.e., the responsibility of TSS, the throughput of a multiplex batch job, and the job about a parallel program.

[0026] The 2nd example (example corresponding to invention according to claim 2 to 4) of the CPU management method of this invention is explained [ 2nd ].

[0027] Drawing 4 is the block diagram showing the composition of the 2nd example of the CPU management method of this invention.

[0028] Compared with the 1st example, as for the CPU management method of this example, the CPU partition activity ratio information CPU partition activity ratio information table 105 and 205 in each back-end cluster 200 in the front end cluster 100 is added.

[0029] Drawing 5 is drawing for explaining concrete operation of the CPU management method of this example.

[0030] Next, operation of the CPU management method of this example constituted in this way is explained. In addition, since it is the same as that of operation in the 1st example about operation other than operation about the CPU partition activity ratio information tables 105 and 205, below, reference is not made.

[0031] By the CPU management method of this example, the activity ratio (activity ratio of CPU belonging to the CPU partition concerned) of a CPU partition is registered into the CPU partition activity ratio information tables 105 and 205. "The activity ratio of a CPU partition" shall be expressed by the time of all CPUs belonging to each CPU partition (CPU partition discriminated by the partition number) in this example. That is, as shown in drawing 5, the CPU partition activity ratio information table 105 (the same is said of CPU partition activity ratio information table 205) has the correspondence information on a "partition number" and a "time." In addition, if the information in the CPU partition activity ratio information table 105 and 205 is information which shows the activity ratio of a CPU partition, it will not be restricted to the information shown in drawing 5.

[0032] As the explanation about the 1st example described, the CPU-time distribution meanses 103 and 203 in each cluster are the process of execution of the multi-cluster system concerned, and after

referring to the CPU partition registration tables 104 and 204, they distribute the CPU time to the generated process. For example, CPU belonging to a CPU partition "3" is assigned to the process which processes a batch job fixed time.

[0033] In this example, the CPU-time distribution means 103 and 203 add the value of the CPU time assigned to the item of the "time" in the entry (entry which has the partition number of the CPU partition with which assigned CPU belongs) to which it corresponds in the CPU partition activity ratio information table 105 and 205 immediately after allocation operation of the CPU time to the above processes further. In an above-mentioned example, the value of the CPU time assigned to the time value in the "time" in the entry of a partition number "3" is added.

[0034] The CPU partition SG means 102 changes dynamically allocation (group division of CPU) of CPU to the CPU partition in each cluster using the CPU partition activity ratio information tables 105 and 205 (optimization).

[0035] Here, in the multi-cluster system concerned, the case where the number of CPU which belongs to the CPU partition "3" in the front end cluster 100 for the purpose of aiming at improvement in the throughput of the multiplex batch job in the front end cluster 100 is made to increase dynamically is considered.

[0036] In this case, in order to increase the number of CPU belonging to the CPU partition "3" in the front end cluster 100 in drawing 2, the directions which show the purport of the increase are performed from a system administrator through the CPU partition input means 101. In addition, these directions are due to the warning (warning which shows that the time of a CPU partition "3" is excessive) which the CPU-time distribution means 103 outputs based on the CPU partition activity ratio information table 105. It is carried out by the system administrator (each of the CPU-time distribution means 103 and the CPU-time distribution means 203). Based on the CPU partition activity ratio information tables 105 or 205 in a self-cluster (cluster in which self exists), an activity ratio can output the warning about the CPU partition which is not suitable.

[0037] Based on the directions, with reference to the CPU operating condition shown on the CPU partition activity ratio information table 105, an activity ratio is vacant as for the CPU partition SG means 102 from a low CPU partition "2", and it moves CPU "20" which is CPU to a CPU partition "3", and changes the CPU partition registration table 104 into the state of drawing 5 from the state of drawing 2.

[0038] Thereby, optimization of the use efficiency of a CPU partition can be attained and improvement in the throughput of a multiplex batch job can be realized as it is this case.

[0039]

[Effect of the Invention] As explained above, this invention prepares a CPU partition input means, a CPU partition SG means, a CPU-time distribution means, and a CPU partition registration table in a multi-cluster system. By carrying out the grouping of the CPU in each cluster, and managing as a CPU partition The CPU time can be assigned so that the character (the purpose of use, i.e., the content of a job) of a process may be suited. It has the effect that improvement in performances, such as TAT of the various performances in a multi-cluster system, i.e., the responsibility of TSS, the throughput of a multiplex batch job, and the job about a parallel program, can be attained.

[0040] Moreover, it is effective in the ability to aim at improvement in the use efficiency of CPU further, being able to adjust the activity ratio of a CPU partition, and allocation of the CPU time of pro SESUHE, and investigating improvement in various performances by preparing a CPU partition activity ratio information table, in addition to the above.

---

[Translation done.]

**\* NOTICES \***

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

**DESCRIPTION OF DRAWINGS**

---

**[Brief Description of the Drawings]**

[Drawing 1] It is the block diagram showing the composition of the 1st example of the CPU management method of this invention.

[Drawing 2] It is drawing for explaining concrete operation of the CPU management method shown in drawing 1.

[Drawing 3] It is the flow chart showing processing of the CPU management method shown in drawing 1.

[Drawing 4] It is the block diagram showing the composition of the 2nd example of the CPU management method of this invention.

[Drawing 5] It is drawing for explaining concrete operation of the CPU management method shown in drawing 4.

**[Description of Notations]**

101 CPU Partition Input Means

102 CPU Partition SG Means

103,203 CPU-time distribution means

104,204 CPU partition registration table

105,205 CPU partition activity ratio information table

---

[Translation done.]

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-091257

(43)Date of publication of application : 04.04.1997

(51)Int.Cl. G06F 15/16  
G06F 9/46

(21)Application number : 07-270513

(71)Applicant : NEC CORP

(22)Date of filing : 25.09.1995

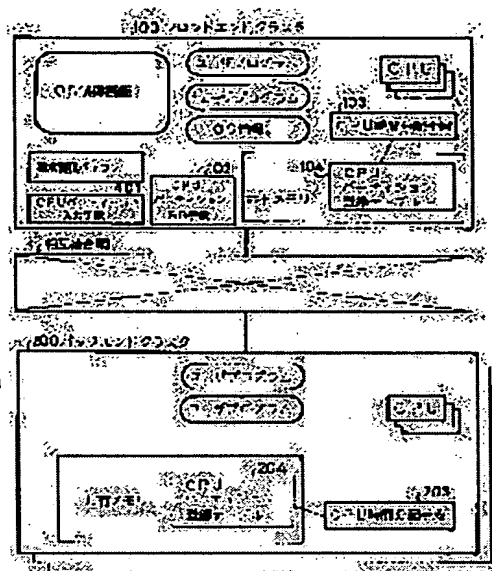
(72)Inventor : TANAKA HIDETOSHI

## (54) CPU MANAGEMENT SYSTEM

## (57)Abstract:

PROBLEM TO BE SOLVED: To optimize the use of CPUs by dividing CPUs into groups to manage them as CPU partitions in a multi-cluster system.

SOLUTION: A CPU partition input means 101 inputs CPU partition information related to respective clusters (a front end cluster 100 and back end clusters 200). A CPU partition SG means 102 divides CPU groups of respective clusters into groups based on CPU partition information and registers information indicating group division in CPU partition registration tables 104 and 204 in each cluster. At the time of the occurrence of a process, a CPU time distribution means 103 or 203 assigns the CPU time related to CPUs belonging to the CPU partition, which is adapted to the character of the process, to the process based on reference to the CPU partition registration table 104 or 204.



## LEGAL STATUS

[Date of request for examination] 25.09.1995

[Date of sending the examiner's decision of rejection] 14.10.1998

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2000 Japan Patent Office



(19) 日本国特許庁 (J P)

## (12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-91257

(43) 公開日 平成9年(1997)4月4日

(51) Int.Cl. <sup>8</sup>	識別記号	序内整理番号	F I	技術表示箇所
G 0 6 F 15/16	3 8 0		G 0 6 F 15/16	3 8 0 Z
9/46	3 6 0		9/46	3 6 0 B

審査請求 有 請求項の数4 F D (全 10 頁)

(21) 出願番号 特願平7-270513

(22) 出願日 平成7年(1995)9月25日

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72) 発明者 田中 英俊

東京都港区芝五丁目7番1号 日本電気株式会社内

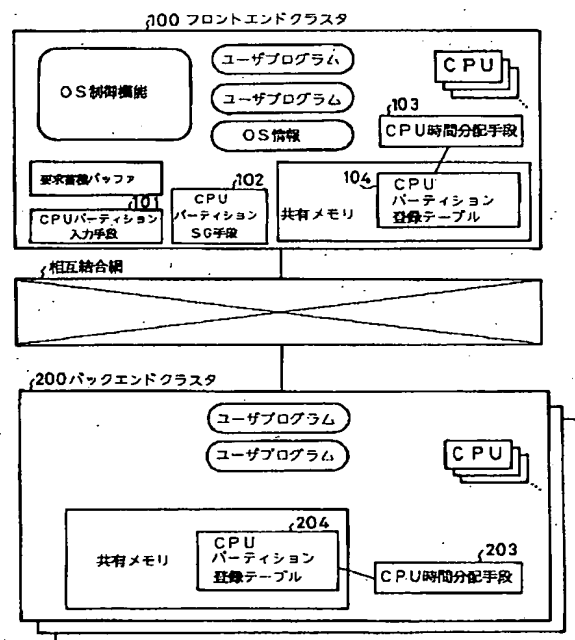
(74) 代理人 弁理士 河原 純一

## (54) 【発明の名称】 CPU管理方式

## (57) 【要約】

【課題】 マルチクラスタシステムにおいて、CPUをグループ化してCPUパーティションとして管理し、CPUの利用の最適化を図る。

【解決手段】 CPUパーティション入力手段101は、各クラスタ（フロントエンドクラスタ100およびバックエンドクラスタ200の各々）に関するCPUパーティション情報を入力する。CPUパーティションSG手段102は、CPUパーティション情報に基づき各クラスタのCPU群のグループ分けを行い、そのグループ分けを示す情報を各クラスタ内のCPUパーティション登録テーブル104および204に登録する。CPU時間分配手段103または203は、プロセスの発生時に、CPUパーティション登録テーブル104または204の参照に基づき、当該プロセスの性格に適合するCPUパーティションに属するCPUに係るCPU時間を当該プロセスに割り当てる。



## 【特許請求の範囲】

【請求項1】 1つのフロントエンドクラスタを含む複数のクラスタを相互結合網で接続したマルチクラスタシステムにおけるCPU管理方式において、各クラスタに関するCPUパーティション情報を入力するフロントエンドクラスタ内のCPUパーティション入力手段と、

前記CPUパーティション入力手段により入力されたCPUパーティション情報に基づいて各クラスタにおける複数のCPUのグループ分けを行い、そのグループ分けを示す情報を各クラスタ内のCPUパーティション登録テーブルに登録するフロントエンドクラスタ内のCPUパーティションSG手段と、プロセスの発生時に前記CPUパーティション登録テーブルの参照に基づいて当該プロセスの性格に適合するCPUパーティションに属するCPUに係るCPU時間を当該プロセスに割り当てる各クラスタ内のCPU時間分配手段とを有することを特徴とするCPU管理方式。

【請求項2】 自クラスタにおける各CPUパーティションの使用率を自クラスタ内のCPUパーティション使用率情報テーブルに登録する各クラスタ内のCPU時間分配手段と、

前記CPUパーティション使用率情報テーブルに基づいて使用率の低いCPUパーティションに属する空きCPUを使用率の高い別のCPUパーティションに動的に割り当て、各CPUパーティションの使用率の最適化を図るCPUパーティションSG手段とを有することを特徴とする請求項1記載のCPU管理方式。

【請求項3】 CPUパーティション使用率情報テーブルが「パーティション番号」と「使用時間」との対応情報を有することを特徴とする請求項2記載のCPU管理方式。

【請求項4】 自クラスタ内のCPUパーティション使用率情報テーブルに基づいて使用率が適切でないCPUパーティションに関する警告を出力する各クラスタ内のCPU時間分配手段を有することを特徴とする請求項2または3記載のCPU管理方式。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 本発明は、マルチクラスタシステムにおけるCPU (Central Processing Unit) 管理方式に関する。

【0002】 ここで、本願明細書でいう「マルチクラスタシステム」とは、以下の①～④に示すような特徴を有するシステムをいう。

① 複数のクラスタ (複数のCPUとそれらの共有メモリとから構成される独立した単位) によって構成される。

② OS (Operating System) 制御機能、OS情報、およびユーザプログラムを有し (これら

の情報は共有メモリに格納されている)、バックエンドクラスタからの要求を蓄積する要求蓄積バッファを持ち、マルチクラスタ環境の中に1つだけ存在可能なフロントエンドクラスタを有する。

③ OS制御機能を持たずに、ユーザプログラムを有し (ユーザプログラムの情報は共有メモリに格納されている)、マルチクラスタ環境に複数個存在可能なバックエンドクラスタを有する。

④ 複数のクラスタ (1つのフロントエンドクラスタおよび1以上のバックエンドクラスタ) が相互結合網 (一般的には高速ネットワーク) で接続されている。

## 【0003】

【従来の技術】 従来より、複数のCPUを有するコンピュータシステムにおいては、CPUをグループ化して、グループに関する情報に基づいてCPUの管理を行うCPU管理方式が存在した (例えば、特開平1-109464号公報参照)。

【0004】 しかし、従来のCPU管理方式では、上述のようなマルチクラスタシステムに関しては、対処することができなかった (上述の特開平1-109464号公報においてもマルチクラスタシステムについては言及されていない)。すなわち、マルチクラスタシステムにおける従来のCPU管理方式では、グループ化の考え方は導入されていなかった。

## 【0005】

【発明が解決しようとする課題】 上述したように、従来のCPU管理方式では、マルチクラスタシステムに適切に対処することができず、CPUをグループ化することによる効果 (各種性能の向上) をマルチクラスタシステムにおいて実現することができないという問題点があった。

【0006】 本発明の目的は、上述の点に鑑み、マルチクラスタシステムにおいて、使用目的 (ジョブ内容) 毎にCPUをグルーピング (グループ化) してCPUパーティションとして管理することにより、CPUの利用の最適化を図り、当該マルチクラスタシステムにおいてCPUのグループ化による各種性能 (マルチクラスタシステムにおけるTSS (Time Sharing System) の応答性、多重バッチジョブのスループット、および並列プログラムに関するジョブのTAT (Turn Around Time) 等の性能) の向上を達成することができるCPU管理方式を提供することにある。

## 【0007】

【課題を解決するための手段】 本発明のCPU管理方式は、1つのフロントエンドクラスタを含む複数のクラスタを相互結合網で接続したマルチクラスタシステムにおけるCPU管理方式において、各クラスタに関するCPUパーティション情報を入力するフロントエンドクラスタ内のCPUパーティション入力手段と、前記CPUパ

10

20

30

40

50

ーティション入力手段により入力されたCPUパーティション情報に基づいて各クラスタにおける複数のCPUのグループ分けを行い、そのグループ分けを示す情報を各クラスタ内のCPUパーティション登録テーブルに登録するフロントエンドクラスタ内のCPUパーティションSG手段と、プロセスの発生時に前記CPUパーティション登録テーブルの参照に基づいて当該プロセスの性格に適合するCPUパーティションに属するCPUに係るCPU時間を当該プロセスに割り当てる各クラスタ内のCPU時間分配手段とを有する。

【0008】

【発明の実施の形態】次に、本発明について図面を参照して詳細に説明する。

【0009】第1に、本発明のCPU管理方式の第1の実施例（請求項1記載の発明に対応する実施例）について説明する。

【0010】図1は、本発明のCPU管理方式の第1の実施例の構成を示すブロック図である。

【0011】本実施例のCPU管理方式は、1つのフロントエンドクラスタ100と複数のバックエンドクラスタ200とが相互結合網（高速ネットワーク）で接続されているマルチクラスタシステムにおいて実現される。

【0012】フロントエンドクラスタ100は、OS制御機能、OS情報、およびユーザプログラム（図示していないが、これらの情報は共有メモリに格納されている）や、要求蓄積バッファ、複数のCPU、および共有メモリを備えた上で、CPUパーティション入力手段101と、CPUパーティションSG（System Generation）手段102と、CPU時間分配手段103と、CPUパーティション登録テーブル104（共有メモリ内に存在する）とを含んで構成されている。

【0013】バックエンドクラスタ200は、複数のCPU、ユーザプログラム、および共有メモリを備えた上で、CPU時間分配手段203と、CPUパーティション登録テーブル204（共有メモリ内に存在する）とを含んで構成されている。

【0014】本実施例のCPU管理方式は、図1に示すようなマルチクラスタシステム内の各クラスタにおいて、複数のCPUを使用目的（ジョブ内容）毎にグルーピングしてCPUパーティションとして管理する。すなわち、各クラスタ（フロントエンドクラスタ100およびバックエンドクラスタ200の各々）における複数のCPUをグルーピングしてそのグループ分けを示す情報を各クラスタに対応するCPUパーティション登録テーブル104または204に登録することにより、CPUパーティションを特定の使用目的（ジョブ内容）毎に管理できるようにする。

【0015】図2は、本実施例のCPU管理方式の具体的な動作を説明するための図である。

【0016】図3は、本実施例のCPU管理方式の処理を示す流れ図である。この処理は、CPUパーティション情報入力ステップ301と、CPUパーティション登録テーブル作成ステップ302と、プロセス発生判定ステップ303と、CPU選択ステップ304と、CPU時間割当てステップ305とからなる。

【0017】次に、このように構成された本実施例のCPU管理方式の動作について説明する。

【0018】フロントエンドクラスタ100内のCPUパーティション入力手段101は、システム管理者からの指示に基づき、CPUパーティションの性格付けを行いその管理を行うための情報であるCPUパーティション情報を、各クラスタ（フロントエンドクラスタ100およびバックエンドクラスタ200の各々）について入力する（ステップ301）。ここで、本実施例におけるCPUパーティション情報とは、具体的には、ジョブ内容（CPUバウンド（bound）ジョブ、インタラクティブ（interactive）ジョブ、およびバッチジョブ等）によって分類されたCPUパーティションと、そのCPUパーティションに属するCPUの個数との、対応情報をいう。

【0019】CPUパーティションSG手段102は、ステップ301で入力されたCPUパーティション情報に基づき、各クラスタにおけるCPUパーティション登録テーブル104および204を作成する（ステップ302）。

【0020】図2の例におけるCPUパーティション登録テーブル104では、CPU「3」、「5」、および「12」（CPUの識別情報が「3」、「5」、および「12」であるCPU）が1つのグループとされ、それらの識別情報がCPUパーティション「1」（CPUバウンドパーティション）に対応する情報として格納されている。また、CPU「17」、「18」、「19」、および「20」が1つのグループとされ、それらの識別情報がCPUパーティション「2」（インタラクティブパーティション）に対応する情報として格納されている。さらに、CPU「7」、「9」、「22」、および「23」が1つのグループとされ、それらの識別情報がCPUパーティション「3」（バッチジョブパーティション）に対応する情報として格納されている。このような情報の格納（登録）が、CPUパーティション登録テーブル104ばかりでなく、各バックエンドクラスタ200内のCPUパーティション登録テーブル204についても行われる。これにより、静的に、CPUの使用目的（ジョブ内容）毎にCPUの識別情報がグループ化されることになる。

【0021】フロントエンドクラスタ100内のCPU時間分配手段103または各バックエンドクラスタ200内のCPU時間分配手段203は、当該マルチクラスタシステムの実行時において、プロセスが生じると（ス

5

テップ303参照)、そのプロセスへのCPU時間の割当てを、以下の①および②に示すようにして行う。

【0022】① CPUパーティション登録テーブル104内のCPUパーティションに関する情報に基づいて、割当て対象のプロセスの性格(使用目的、すなわち当該プロセスに関するジョブ内容)に適合したCPUパーティションに属する空きCPUの1つを選択する(ステップ304)。

【0023】② ①で選択したCPUを一定時間そのプロセスに割り当て、すなわちそのCPUに係るCPU時間をそのプロセスに割り当て(ステップ305)、その後10にステップ303の判定を行う状態に戻る。

【0024】以上のようにして、当該マルチクラスタシステム内の多数のCPUをグルーピングしてCPUパーティションとして管理することができるようになる。

【0025】このようなCPUパーティション(CPUのグループ化)の考え方を導入することにより、マルチクラスタシステムにおける各種性能、すなわちTSSの応答性、多重バッチジョブのスループット、および並列プログラムに関するジョブのTAT等の性能に関して、20CPU時間の分配の最適化を図ることができる。

【0026】第2に、本発明のCPU管理方式の第2の実施例(請求項2〜4記載の発明に対応する実施例)について説明する。

【0027】図4は、本発明のCPU管理方式の第2の実施例の構成を示すブロック図である。

【0028】本実施例のCPU管理方式は、第1の実施例に比べて、フロントエンドクラスタ100内のCPUパーティション使用率情報テーブル105および各バックエンドクラスタ200内のCPUパーティション使用率情報テーブル205が追加されている。30

【0029】図5は、本実施例のCPU管理方式の具体的な動作を説明するための図である。

【0030】次に、このように構成された本実施例のCPU管理方式の動作について説明する。なお、CPUパーティション使用率情報テーブル105および205に関する動作以外の動作については第1の実施例における動作と同様であるので、以下では言及しない。

【0031】本実施例のCPU管理方式では、CPUパーティションの使用率(当該CPUパーティションに属するCPUの使用率)がCPUパーティション使用率情報テーブル105および205に登録される。本実施例では、各CPUパーティション(パーティション番号で識別されるCPUパーティション)に属する全CPUの使用時間によって「CPUパーティションの使用率」が表現されるものとする。すなわち、図5に示すように、CPUパーティション使用率情報テーブル105(CPUパーティション使用率情報テーブル205も同様)は、「パーティション番号」と「使用時間」との対応情報を有している。なお、CPUパーティション使用率情50

6

報テーブル105および205内の情報は、CPUパーティションの使用率を示す情報であれば、図5に示す情報に限られるものではない。

【0032】第1の実施例に関する説明で述べたように、各クラスタにおけるCPU時間分配手段103および203は、当該マルチクラスタシステムの実行の過程で、CPUパーティション登録テーブル104および204を参照した上で、発生したプロセスにCPU時間を振り分ける。例えば、バッチジョブの処理を行うプロセスにはCPUパーティション「3」に属するCPUを一定時間割り当てる。

【0033】本実施例では、さらに、CPU時間分配手段103および203は、上述のようなプロセスへのCPU時間の割当て動作の直後に、CPUパーティション使用率情報テーブル105および205内の該当するエントリ(割り当てられたCPUが属するCPUパーティションのパーティション番号を有するエントリ)中の「使用時間」の項目に割り当てたCPU時間の値を加算する。上述の例では、パーティション番号「3」のエントリ中の「使用時間」における時間値に割り当てたCPU時間の値を加算する。

【0034】CPUパーティションSG手段102は、CPUパーティション使用率情報テーブル105および205を利用して、各クラスタにおけるCPUパーティションへのCPUの割当て、(CPUのグループ分け)を動的に変更(最適化)する。

【0035】ここでは、当該マルチクラスタシステムにおいて、フロントエンドクラスタ100における多重バッチジョブのスループットの向上を図ることを目的として、フロントエンドクラスタ100におけるCPUパーティション「3」に属するCPUの個数を動的に増加させる場合について考える。

【0036】この場合には、図2におけるフロントエンドクラスタ100におけるCPUパーティション「3」に属するCPUの個数を増やすために、その増加の旨を示す指示がCPUパーティション入力手段101を介してシステム管理者から行われる。なお、この指示は、例えば、CPUパーティション使用率情報テーブル105に基づいてCPU時間分配手段103が出力する警告(CPUパーティション「3」の使用時間が過大であることを示す警告)に基づいて、システム管理者によって行われる(CPU時間分配手段103およびCPU時間分配手段203の各々は、自クラスタ(自己が存在するクラスタ)内のCPUパーティション使用率情報テーブル105または205に基づいて、使用率が適切でないCPUパーティションに関する警告を出力することができる)。

【0037】CPUパーティションSG手段102は、その指示に基づいて、CPUパーティション使用率情報テーブル105によって示されるCPU使用状況を参照

して、使用率が低いCPUパーティション「2」から空きCPUであるCPU「20」をCPUパーティション「3」に移動し、CPUパーティション登録テーブル104を図2の状態から図5の状態に変更する。

【0038】これにより、CPUパーティションの使用効率の最適化を図ることができ、この場合であると、多重バッチジョブのスループットの向上を実現することができる。

【0039】

【発明の効果】以上説明したように本発明は、マルチクラスタシステムにおいて、CPUパーティション入力手段、CPUパーティションSG手段、CPU時間分配手段、およびCPUパーティション登録テーブルを設け、各クラスタにおいてCPUをグルーピングしてCPUパーティションとして管理することにより、プロセスの性格（使用目的、すなわちジョブ内容）に適合するようにCPU時間の割当てを行うことができ、マルチクラスタシステムにおける各種性能、すなわちTSSの応答性、多重バッチジョブのスループット、および並列プログラムに関するジョブのTAT等の性能の向上を達成することができるという効果を有する。

【0040】また、上記に加えて、CPUパーティション使用率情報テーブルを設けることにより、CPUパー

ティションの使用率とプロセスへのCPU時間の割当てとを調整することができ、各種性能の向上を追及しつつ、CPUの使用効率の向上をさらに図ることができるという効果がある。

【図面の簡単な説明】

【図1】本発明のCPU管理方式の第1の実施例の構成を示すブロック図である。

【図2】図1に示すCPU管理方式の具体的な動作を説明するための図である。

【図3】図1に示すCPU管理方式の処理を示す流れ図である。

【図4】本発明のCPU管理方式の第2の実施例の構成を示すブロック図である。

【図5】図4に示すCPU管理方式の具体的な動作を説明するための図である。

【符号の説明】

101 CPUパーティション入力手段

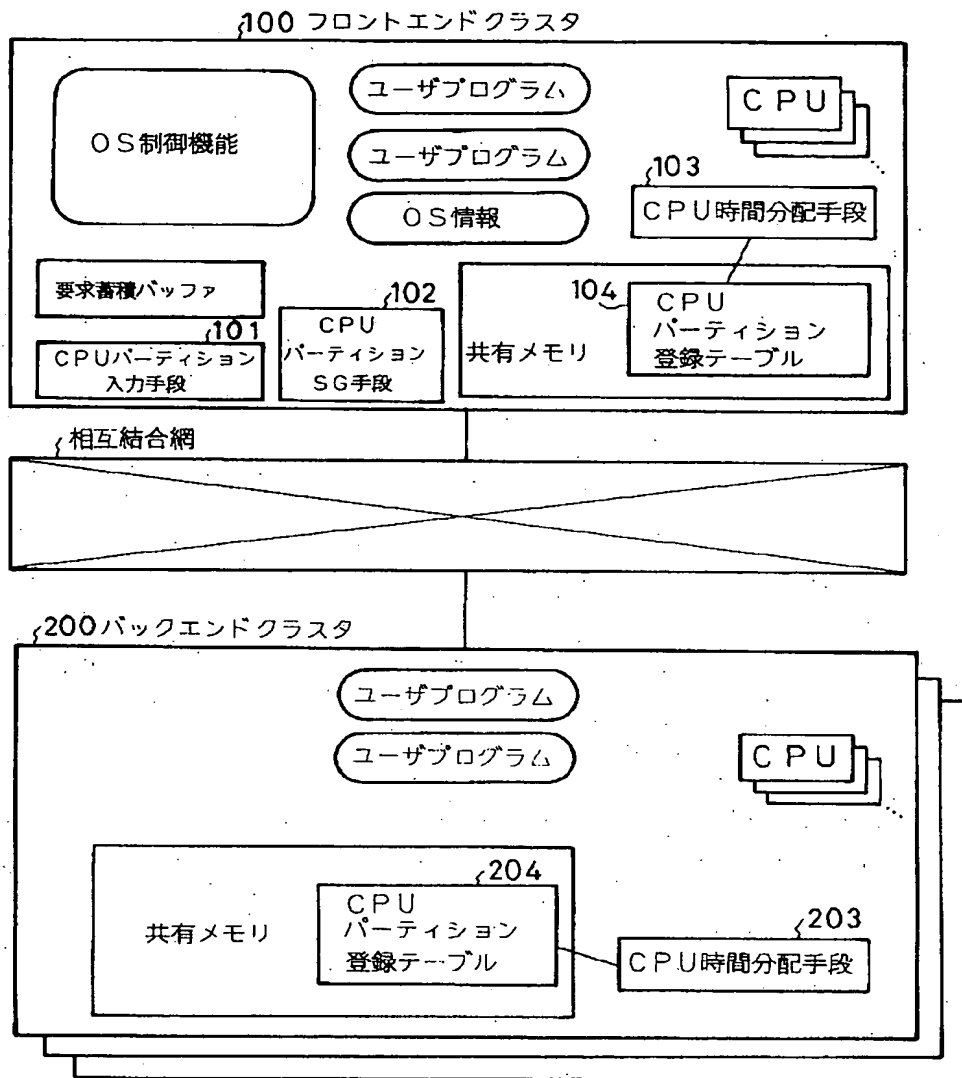
102 CPUパーティションSG手段

103, 203 CPU時間分配手段

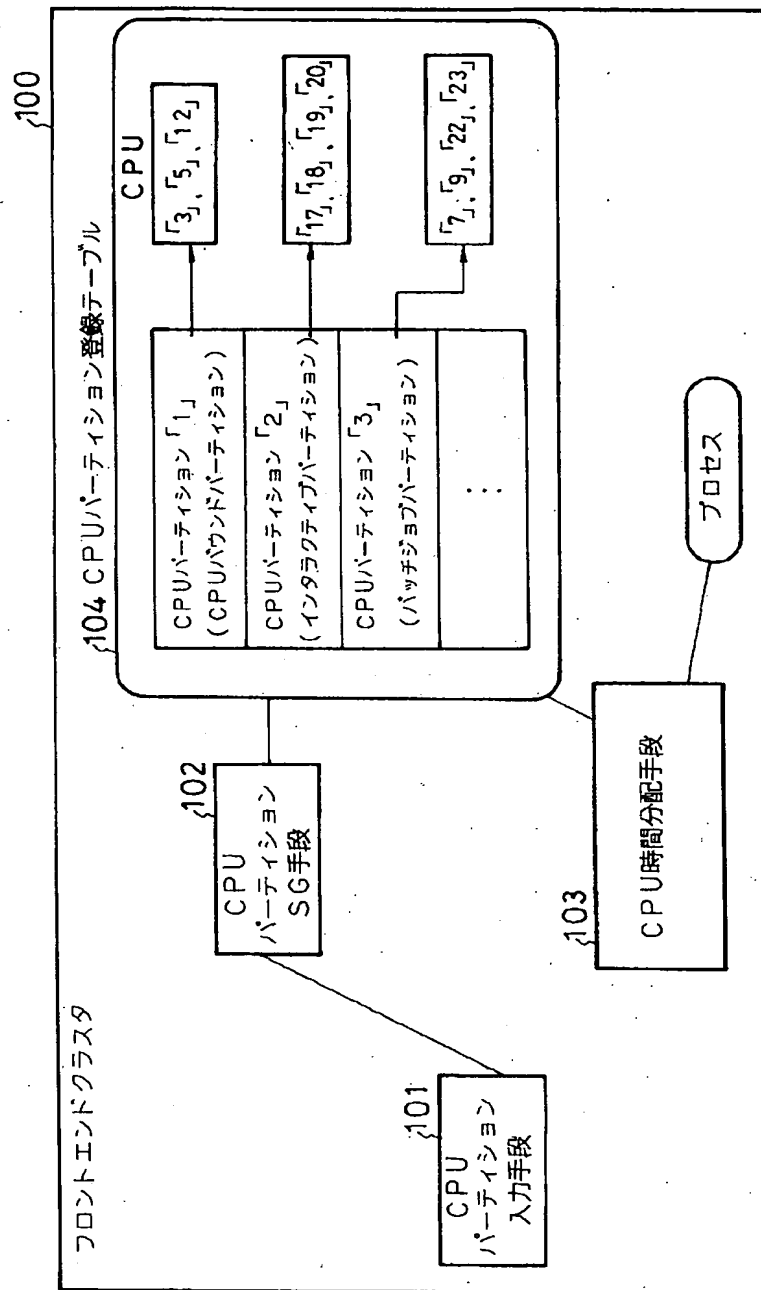
104, 204 CPUパーティション登録テーブル

105, 205 CPUパーティション使用率情報テーブル

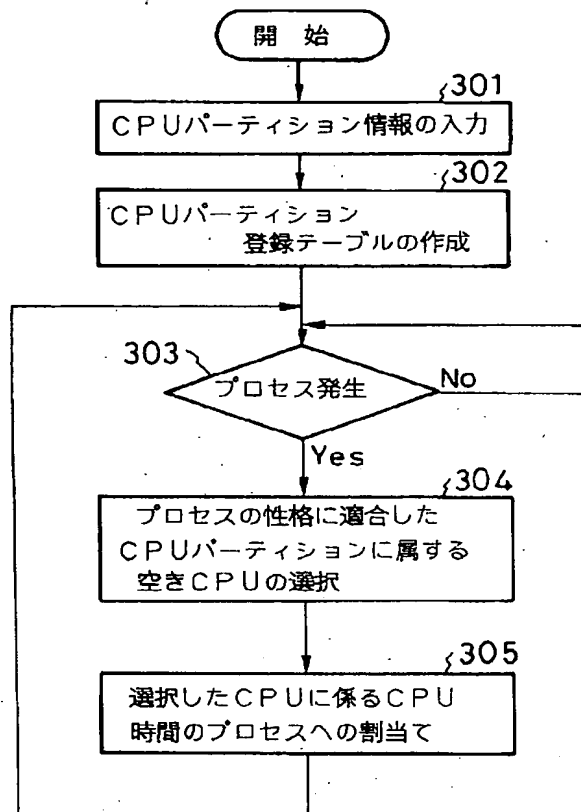
【図1】



【図2】

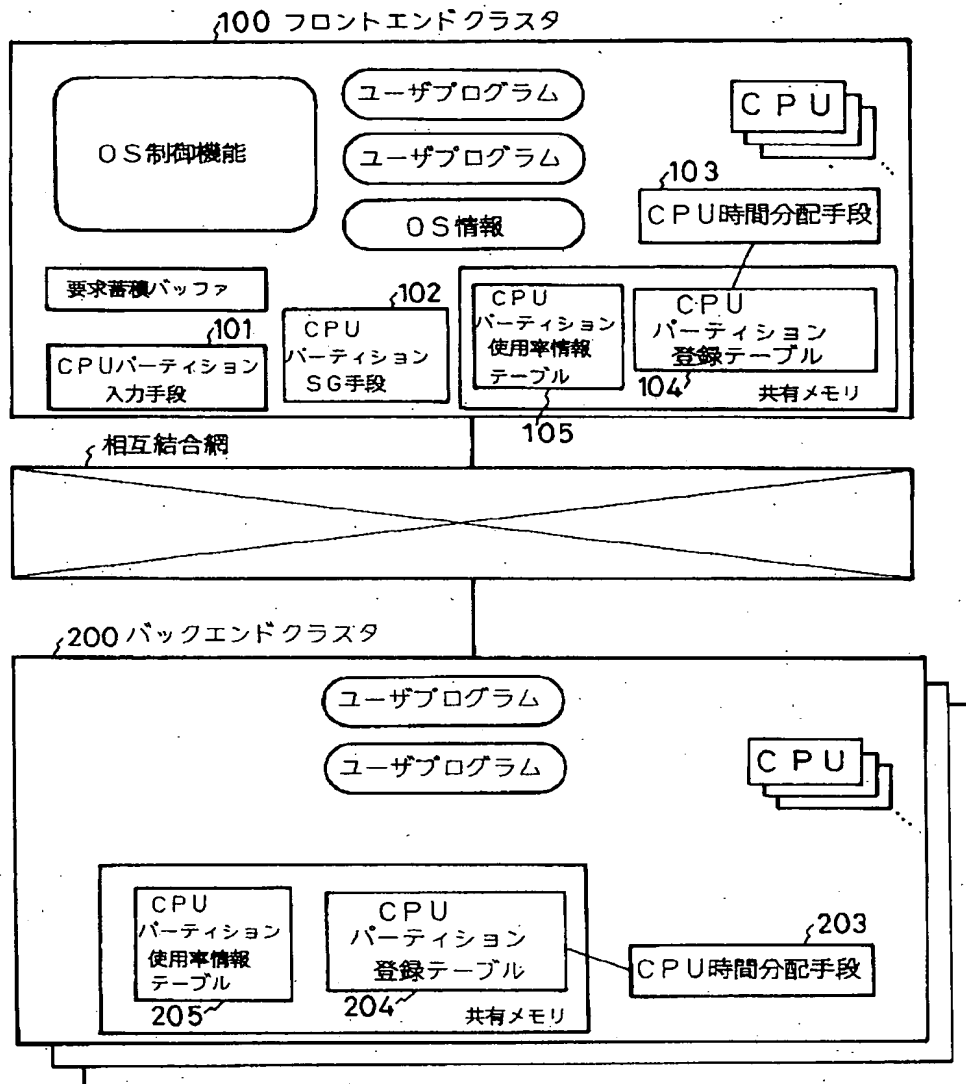


【図3】





【図4】



【図 5】

